TEXT GENERATION

1.1-INTRODUCTION

What is PYTHON ?

Python is a popular , interpreted, high-level and a multi-paradigm programming language. It was created by Guido van Rossum and released in 1991. Object-oriented programming and structured programming are fully supported.

It is used for :

* Web development (server-side).
* Software development.
* Mathematics.
* System scripting.

Why Python?

* Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
* Python has a simple syntax similar to the English language.
* Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
* Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
* Python can be treated in a procedural way, an object-orientated way or a functional way.

INTRODUCTION TO ARTIFICIAL INTELLIGENCE

The term "artificial intelligence" is used to describe machines that mimic "cognitive" functions that humans associate with other human minds, such as "learning" and "problem solving. In computer science, artificial intelligence (AI), sometimes called machine intelligence (ML) , is intelligence demonstrated by machines, in contrast to the natural intelligence displayed by humans and animals.

Some of the activities computers with artificial intelligence are designed for include:

* Speech recognition.
* Learning.
* Planning.

Artificial intelligence can be classified into three different types of systems:

* Analytical Artificial Intelligence : Analytical AI has only characteristics consistent with cognitive intelligence; generating a cognitive representation of the world and using learning based on past experience to inform future decisions
* Human-inspired Artificial Intelligence : Human-inspired AI has elements from cognitive and emotional intelligence; understanding human emotions, in addition to cognitive elements, and considering them in their decision making
* Humanized Artificial Intelligence : Humanized AI shows characteristics of all types of competencies, and is able to be self-conscious and is self-aware in interactions with others.

..

1.2 OBJECTIVES OF RESEARCH:

Text generation is a subfield of natural language processing. It leverages knowledge in computational linguistics and artificial intelligence to automatically generate natural language texts, which can satisfy certain communicative requirements.

1.3 PROBLEM STATEMENT:

To perform Text Generation with Long Short Term Memory (LSTM),Recurrent Neural Netwoks.

This can be achieved by training the model with the dataset using RNN

2. REVIEW OF LITERATURE:

We used the dataset of a newspaper’s Comments and Headlines to train the text generation language model which is useful to generate News Headlines.

This solution contains six steps :

1. Importing the libraries : We need to import the required libraries such as keras module for building LSTM , NumPy library ( as it provides objects for multi-dimensional arrays ) and Pandas library ( as is provides data analysis tools ).

2. Loading the dataset of news headlines and comments.

3. Dataset preprocessing : In this step, we first perform text cleaning of the data which includes removal of punctuations and lower casing all the words. The next step is Tokenization. Tokenization is a process of extracting terms or words ( tokens ) from a corpus. Python’s library Keras has inbuilt model for tokenization which can be used to obtain the tokens and their index in the corpus. After this step, every text document in the dataset is converted into sequence of tokens.

Before starting training the model, we need to pad the sequences and make their lengths equal. We can use pad\_sequence function of Keras for this purpose. To input this data into the learning model, we need to create predictors and label. We create N-grams sequence as predictors and the next word of the N-gram as label.

4.LSTM for text generation:

1.Input Layer : Takes the sequence of words as input.

2. LSTM Layer : Computes the output using LSTM units.

3. Dropout Layer : It helps in preventing over-fitting and it is a regularization layer which turns off the activations of some neurons in LSTM layer.

4. Output Layer : Computes the probability of the possible next word as output.

5. Generating the text : We will first tokenize the text, pad the sequences and pass into the trained model to get predicted words.

3. DATA COLLECTION

SOURCE: https://www.kaggle.com/aashita/nyt-comments

DATASET: New York Times Comments and Headlines.

This data contains information about the comments made on the articles published in New York Times in Jan-May 2017 and Jan-April 2018. The month-wise data is given in two csv files - one each for the articles on which comments were made and for the comments themselves.

4. METHODOLOGY

The text generation can be predicted using the following parameters .

The methods used in this model are:

1. Data collection
2. Data cleaning
3. Tokenization
4. LSTM model for text generation (Recurrent Neural Networks ).
5. Training the model
6. User interface

1.Data collection is the systematic approach to gathering and measuring information from a variety of sources to get a complete and accurate picture of an area of interest.

2.Data cleaning,in this we brought all the head lines in to one list and removed punctuations later converting every letter in to lower case.

3.Tokenization is a process of extracting terms or words ( tokens ) from a corpus. Python’s library Keras has inbuilt model for tokenization which can be used to obtain the tokens and th is converted into sequence of tokens.

4.Then we created the model using four layers. neurons propagate in both directions (from inputs to outputs and from outputs to inputs) in Recurrent Neural Networks. This creates loops in the neural network architecture which acts as a ‘memory state’ of the neurons. A new type of RNNs called LSTMs (Long Short Term Memory) Models have been developed.

5.We fit the model by number of iterations updating weights each time.

6.User Interface is designed using tkinter.

5. FINDINGS AND IDEAS

As we can see, the model has produced the output which looks fairly fine. The results can be improved further with following points:

* Adding more data
* Fine Tuning the network architecture
* Fine Tuning the network parameters
* User interface can be developed further

6.CONCLUSION

The need of a text generator is undeniable and Grammarly,Google keyboard are a popular example for this.We explored several aspects of AI in the process of developing this project and feel this is just the start of an exciting journey!!!